Clean Version of Pending Claims

HIERARCHICAL SOFTWARE PATH PROFILING
Applicant: Youfeng Wu et al.
Serial No.: 09/541,399

Please replace the previously pending claim set with the claims, as amended, below:

1.(Amended). A computer-implemented method comprising:

after entering an inner region of a hierarchical software path, saving an outer path sum;

initializing an inner path sum;

summing edge values encountered in the inner region with the inner path sum; and

after exiting the inner region, modifying a profile indicator that represents the frequency of execution of a path within the inner region, and restoring the outer path sum.

2. The computer-implemented method of claim 1 wherein initializing an inner path sum comprises initializing the inner path sum to a value corresponding to an edge from a region source node to an entry node of the inner region.

3. The computer-implemented method of claim 2 wherein modifying a profile indicator comprises indexing into an array of profile indicators using the inner path sum.

4. The computer-implemented method of claim 3 wherein the array of profile indicators is dedicated to paths in the inner region.

5. The computer-implemented method of claim 1 wherein initializing an inner path sum comprises initializing the inner path sum to a value corresponding to an edge from a function entry to an entry node of the inner region.

6. The computer-implemented method of claim 5 wherein modifying a profile indicator comprises indexing into an array of profile indicators using the inner path sum.

7.   The computer-implemented method of claim 6 wherein the array of profile indicators includes profile indicators corresponding to paths in the inner region and profile indicators corresponding to paths outside the inner region.

8.   The computer-implemented method of claim 1 wherein the profile indicator includes a profile counter, and modifying the profile indicator comprises generating a counter address as a function of the inner path sum.

9.   The computer-implemented method of claim 1 wherein the inner region includes a plurality of paths, each having an inner path sum corresponding thereto, the inner path sums corresponding to the plurality of paths in the inner region being unique relative to each other.

10.   The computer-implemented method of claim 9 wherein the inner region is one of a plurality of inner regions, and the inner path sums of the inner region are unique relative to inner path sums corresponding to other inner regions.

11.   The computer-implemented method of claim 1 wherein saving an outer path sum comprises pushing the outer path sum onto a stack.

12.   The computer-implemented method of claim 11 wherein restoring the outer path sum comprises popping the outer path sum from the stack.

13.(Amended) A computer-implemented method comprising:

        providing a control flow-graph of a program, the graph having an inner region and an outer region;

        selecting a representative entry node for the inner region;

        replacing the inner region with the representative entry node;

~~for each prolog~~ node of the inner region, adding an edge from the prolog node to the representative entry node; and

for each epilog node of the inner region, adding an edge from the representative entry node to the epilog node.

14.   The computer-implemented method of claim 13 further comprising:

assigning edge values to all edges in the control flow graph such that the sum of the edge values along each unique path is unique within the control flow graph.

15.   The computer-implemented method of claim 13 further comprising:

creating a region source node for the outer region;

for each entry node of the outer region, adding an edge from the region source node to the entry node;

creating a region sink node for the outer region; and

for each exit node of the outer region, adding an edge from the exit node to the region sink node.

16.   The computer-implemented method of claim 15 wherein the control flow graph includes a plurality of inner regions, and the actions of the method are applied for each of the plurality of inner regions, such that a different augmented control flow graph is created for each of the plurality of inner regions.

17.   The computer-implemented method of claim 15 wherein the control flow graph includes a hierarchy of inner regions, and the actions of the method are applied recursively to the hierarchy of inner regions, such that a different augmented control flow graph is created for each inner region in the hierarchy of inner regions.

18.(Amended) A computer-implemented method comprising:

selecting a representative path within an inner region of a software function, the representative path being identified by a representative entry node and a representative exit node;

for each prolog node of the inner region, adding an edge from the prolog node to the representative entry node; and

for each epilog node of the inner region, adding an edge from the representative exit node to the epilog node.

19.   The computer-implemented method of claim 18 further comprising:

removing any edges from prolog nodes of the inner region to entry nodes of the inner region other than the representative entry node; and

removing any edges from exit nodes of the inner region other than the representative exit node to epilog nodes of the inner region.

20.   The computer-implemented method of claim 18 wherein the software function has a function entry and a function exit, and the inner region has at least one entry node and at least one exit node, the method further comprising:

adding an edge from the function entry to each of the at least one entry node of the inner region; and

adding an edge from each of the at least one exit node of the inner region to the function exit.

21.   The computer-implemented method of claim 20 wherein the control flow graph includes a plurality of inner regions, and the actions of the method are applied for each of the plurality of inner regions.

22.   The computer-implemented method of claim 20 wherein the control flow graph includes a hierarchy of inner regions, and the actions of the method are applied recursively to the hierarchy of inner regions.

23.   The computer-implemented method of claim 20 further comprising:

assigning edge values to all edges in the control flow graph such that the sum of the edge values along each unique path is unique within the control flow graph.

24.(Amended) A machine readable medium including instructions for a method comprising:

selecting a representative path within an inner region of a software function, the representative path being identified by a representative entry node and a representative exit node;

for each prolog node of the inner region, adding an edge from the prolog node to the representative entry node; and

for each epilog node of the inner region, adding an edge from the representative exit node to the epilog node.

25.   The machine readable medium of claim 24 wherein the software function has a function entry and a function exit, and the inner region has at least one entry node and at least one exit node, the method further comprising:

adding an edge from the function entry to each of the at least one entry node of the inner region; and

adding an edge from each of the at least one exit node of the inner region to the function exit.

26.(Amended) A computer-implemented method comprising:

at an entry to an inner region of a hierarchical software path, inserting an instruction to save an outer path sum;

within the inner region, inserting instructions to sum edge values into an inner path sum; and

after an exit from the inner region, inserting an instruction to increment a counter addressed as a function of the inner path sum.

27.(Amended) The computer-implemented method of claim 26 further comprising:

after the exit from the inner region, inserting an instruction to restore the outer path sum.

28.   The computer-implemented method of claim 27 wherein the inner region is one of a plurality of inner regions in a hierarchical arrangement, and the actions of the method are applied to each of the plurality of inner regions.

29.(Amended) A machine readable medium including instructions for a method comprising:

at an entry to an inner region of a hierarchical software path, inserting an instruction to save an outer path sum;

within the inner region, inserting instructions to sum edge values into an inner path sum; and

after an exit from the inner region, inserting an instruction to increment a counter addressed as a function of the inner path sum.

30.(Amended) The machine readable medium of claim 29, the method further comprising:

after the exit from the inner region, inserting an instruction to restore the outer path sum.

31. A computer-implemented method comprising:

defining an outer region of code for a computer program;

defining an inner region of the code embedded within the outer region;

assigning inner edge values within the inner region such that the inner edge values have

different sums for different paths through the nodes of the inner region;

assigning outer edge values within the outer region such that the outer edge values have different sums for different paths through the nodes of the outer region, where the inner region is treated as a single node;

while executing the outer region, accumulating an outer path sum of the outer edge values;

in connection with entering the inner region,

saving the outer path sum,

accumulating an inner path sum of the inner edge values;

in connection with exiting the inner region,

modifying a profile indicator that represents the frequency of execution of a path within the inner region,

restoring the outer path sum.

32. The computer-implemented method of claim 31 wherein the code is executable code.

33. The computer-implemented method of claim 31 wherein the inner path sum is uniquely determined for different execution paths in the inner region.

34. The computer-implemented method of claim 33 further comprising addressing the profile indicator with the accumulated inner path sum.

35. The computer-implemented method of claim 31 further comprising initializing the inner path sum before accumulating it.

36. The computer-implemented method of claim 31 further comprising repeating the operations of claim 31 wherein the outer region is an inner region embedded in a further outer region.

37. The computer-implemented method of claim 36 wherein saving an outer path sum comprises pushing the outer path sum onto a stack.

38. The computer-implemented method of claim 36 wherein restoring the outer path sum comprises popping the outer path sum from a stack.

39. A machine-readable medium including instructions to carry out the method of claim 31.